Recent Progress in Post-Processing OVERFLOW-D Simulations

David Edwards
Mike Schechterman
Intelligent Light

Earl Duque
Northern Arizona University

Outline

- On-going research in post-processing OVERFLOW-D simulations
  - Utilization of rotorcraft domain knowledge
  - Particle Integration
- Open discussion

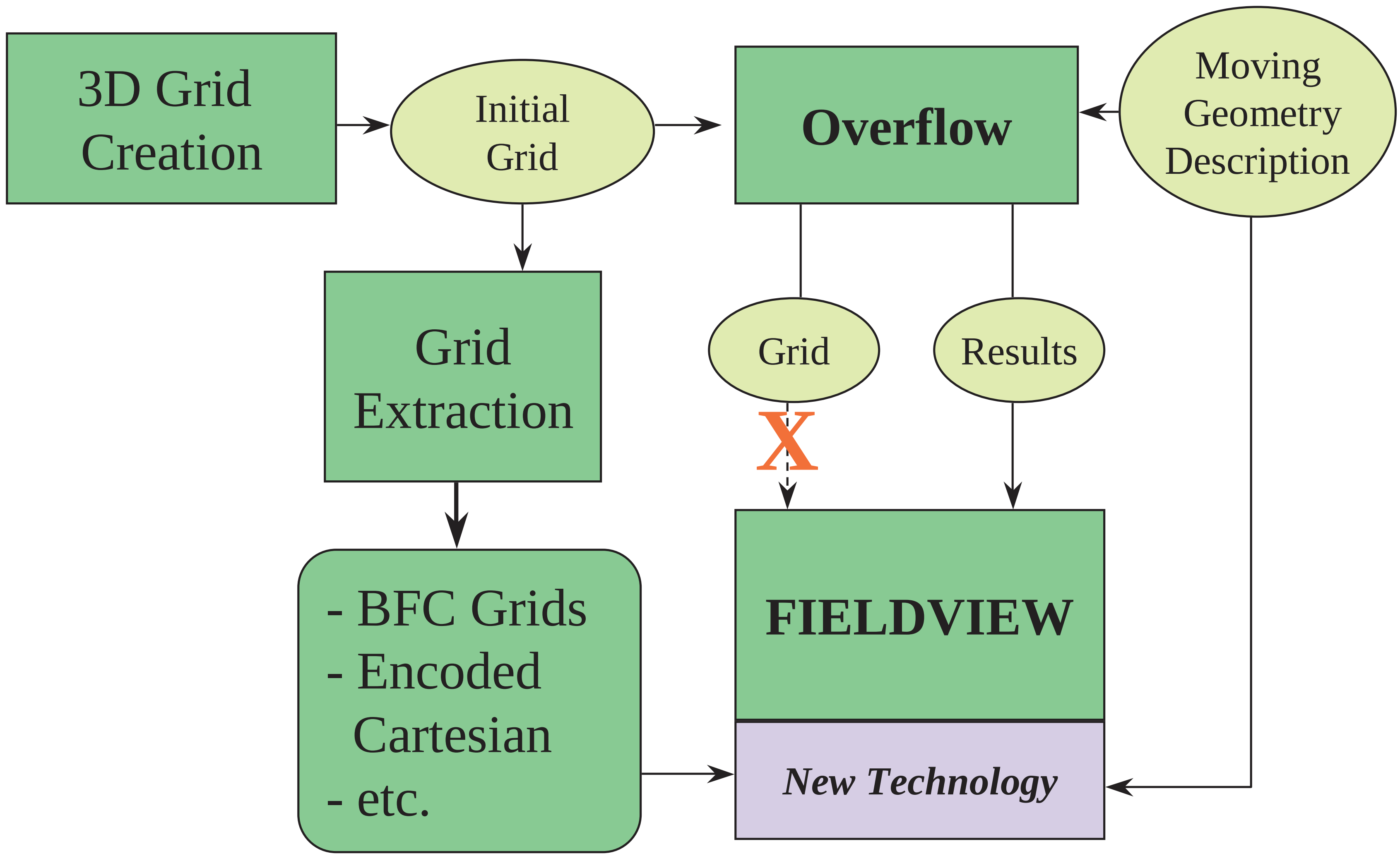
Utilization of Rotorcraft Domain Knowledge to Optimize Process Post-Processing

Goal: Improve performance in terms of computational & memory resources

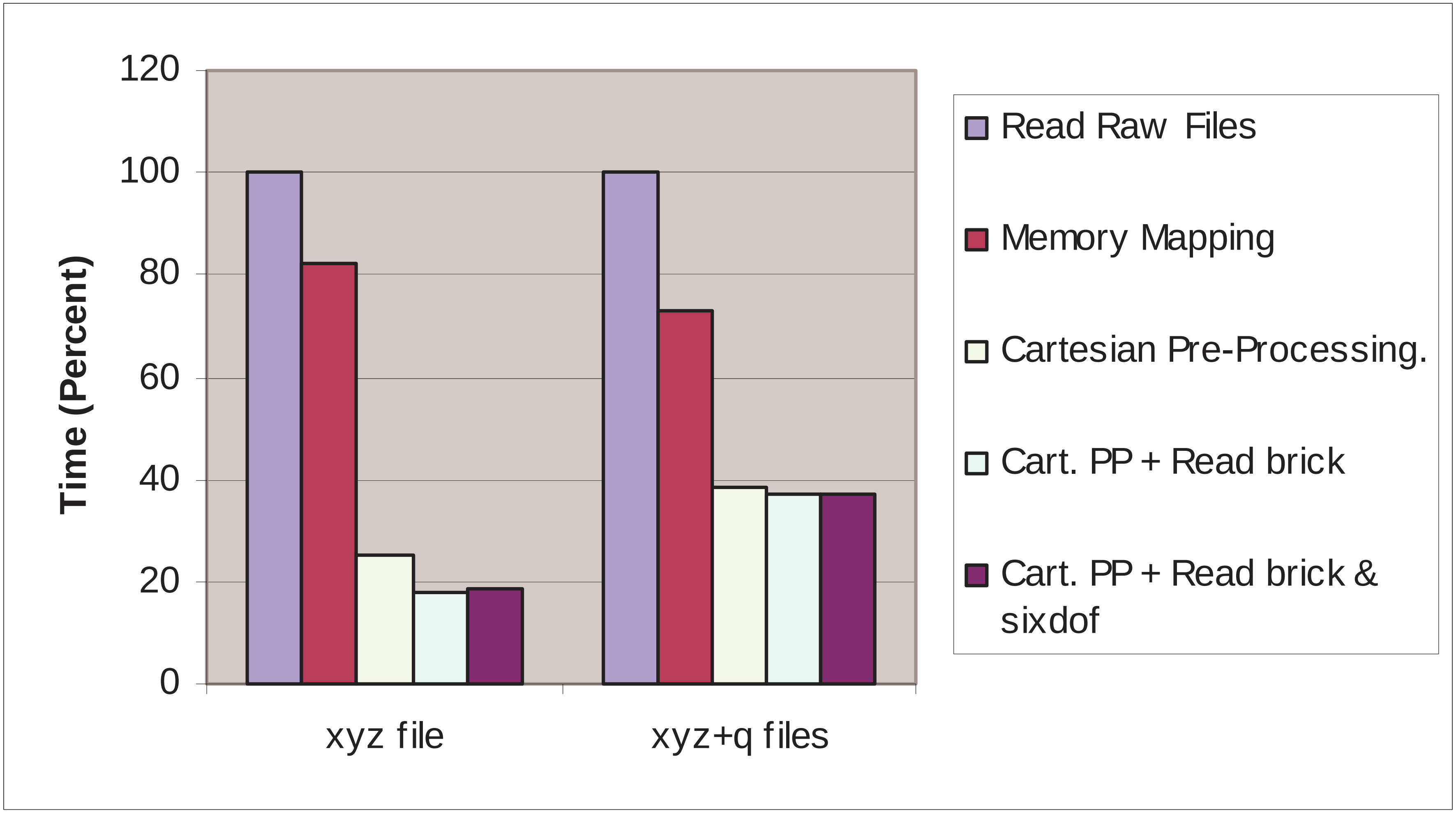
Areas examined:

1. Reduce reading in grids that are Cartesian and construct grids on-the-fly using “header information” to construct the grid.
2. Construct transient rigid-body grids on-the-fly using grid transformations for each transient step instead of reading in grid data at each node point.
3. Minimize start-up operations at each transient step

Proposed Transient Data Flow



Impact of Rotorcraft Domain Knowledge



Utilizing Cartesian and Rotorcraft Knowledge Post Processing allows:

- Grid file size can be reduced substantially
  - 60% for a steady solution
  - ~75 % for a transient solution
  - Allows for designing new post-processing OVERFLOW-D grid file

- Startup time per transient time-step is faster

Motivation for Improvements in Particle Integration

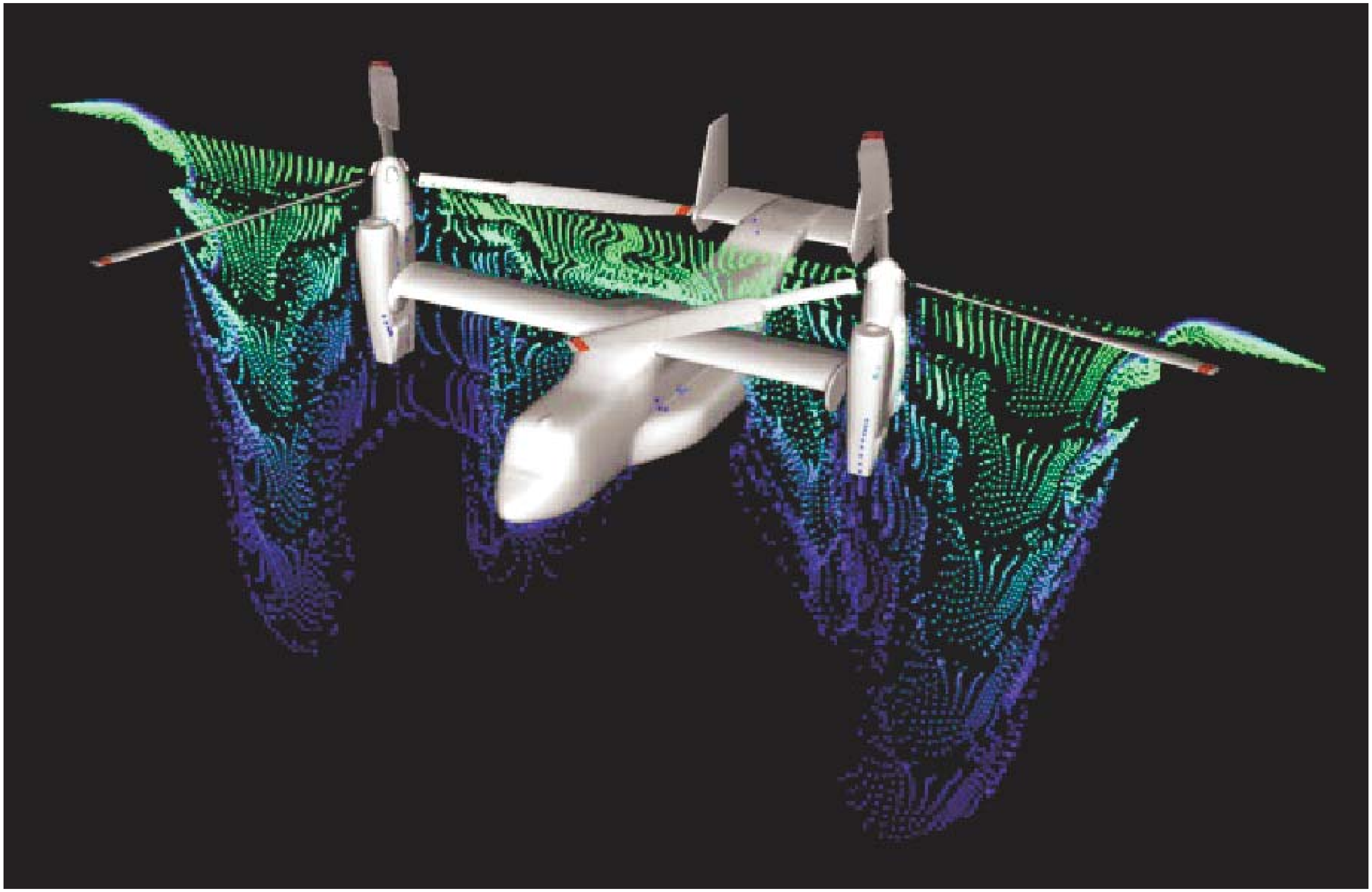


Image courtesy of Mark Potsdam, Army Rotorcraft Aeromechanics Branch, NASA Ames
Streakline Integration - UFAT

Analyze Streakline Calculation In Flows with Moving Bodies

- Review current streakline integration schemes (pV3, GEL, UFAT)
- Using OVERFLOW-D data, benchmark streakline integration scheme (a 4^{th} order implicit backward differentiation BD4 scheme) and indicate deficiencies.
  - Test with fine and coarse time steps
  - Release streaklines near moving walls where problems will occur
- Evaluate need for additional modeling
  - Restriction of streaklines around solid walls
  - Integration scheme approximations

Simulations for Numerical Studies

- Created by Northern Arizona University
  - Generated on NAU Beowulf Cluster
- Two simulations
  - Rotating body
  - Rotor at hover

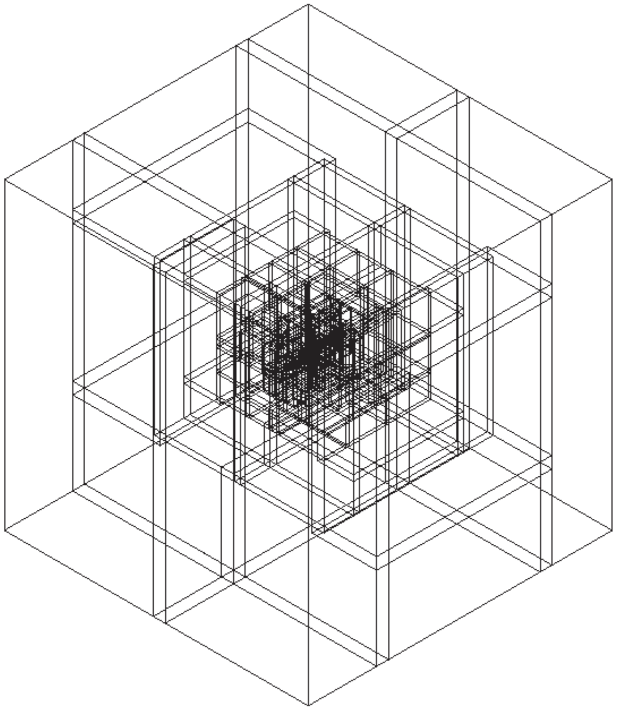
Numerical Studies

- Impact of streakline integration method
  - Backward Difference (BD4)
    - 4th order local error, 3rd order global error
  - Runge-Kutta (RK4)
    - 4th order local error, 3rd order global error

- Impact of Time Step
  - Time steps of 1, 2, 4, 8 degrees

- Impact of Temporal Sub-steps
  - No steps, 10 sub-steps
  - Linear in computational space

Streakline Model Problem 1



- 85 Grids
- 360 Time Steps (1degree rotation per time-step)
- Inlet Mach Number = .2

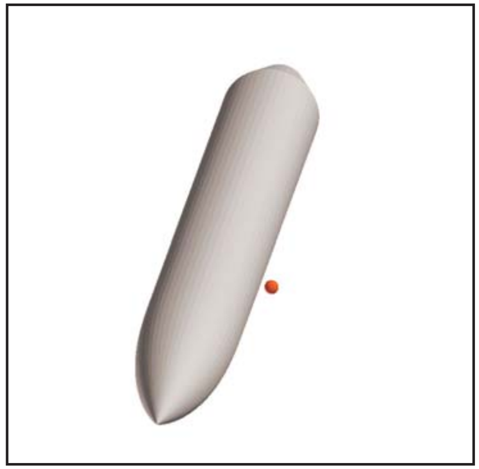
Basics of Numerical Study

- Series of streaklines calculations were performed
  - Integration method
  - Time step
  - Time sub-step
- Compare results to a baseline case (BD4 with 1 degree time-step).
  - Examined streakline at a specific time and determine distance from baseline case at same time
  - Selected six points in field for analysis

Point	Time	X	Y	Z
T0	751.9531	-19.50000	-2.25000	-3.75000
T1	777.5244	-15.35526	-2.67705	-3.05959
T2	841.4523	-1.65766	-1.56723	-4.28803
T3	892.5947	9.10078	0.92924	-2.84787
T4	943.7371	18.57602	-3.86564	-4.06998
T5	994.8795	27.44151	-4.28093	-7.00792
T6	1020.451	31.87707	-4.72556	-7.52175

Location of Reference Points

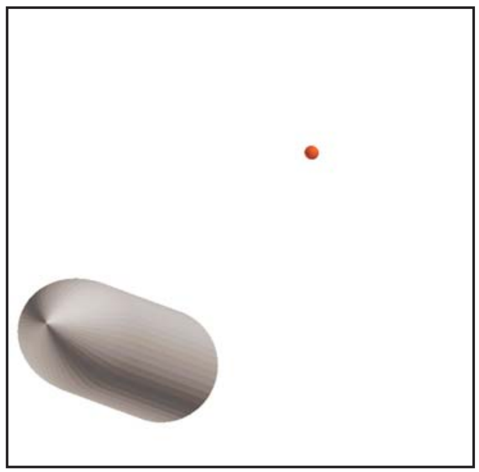
Point T1



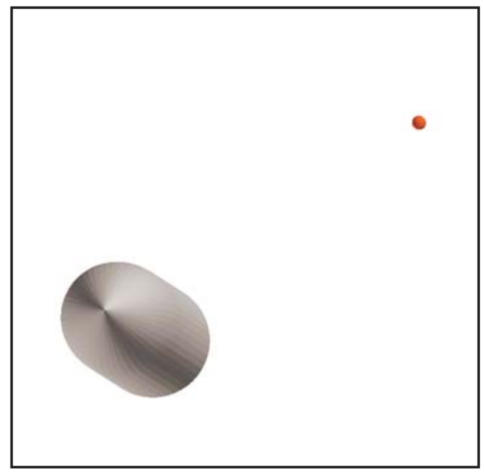
Point T2

Point T3

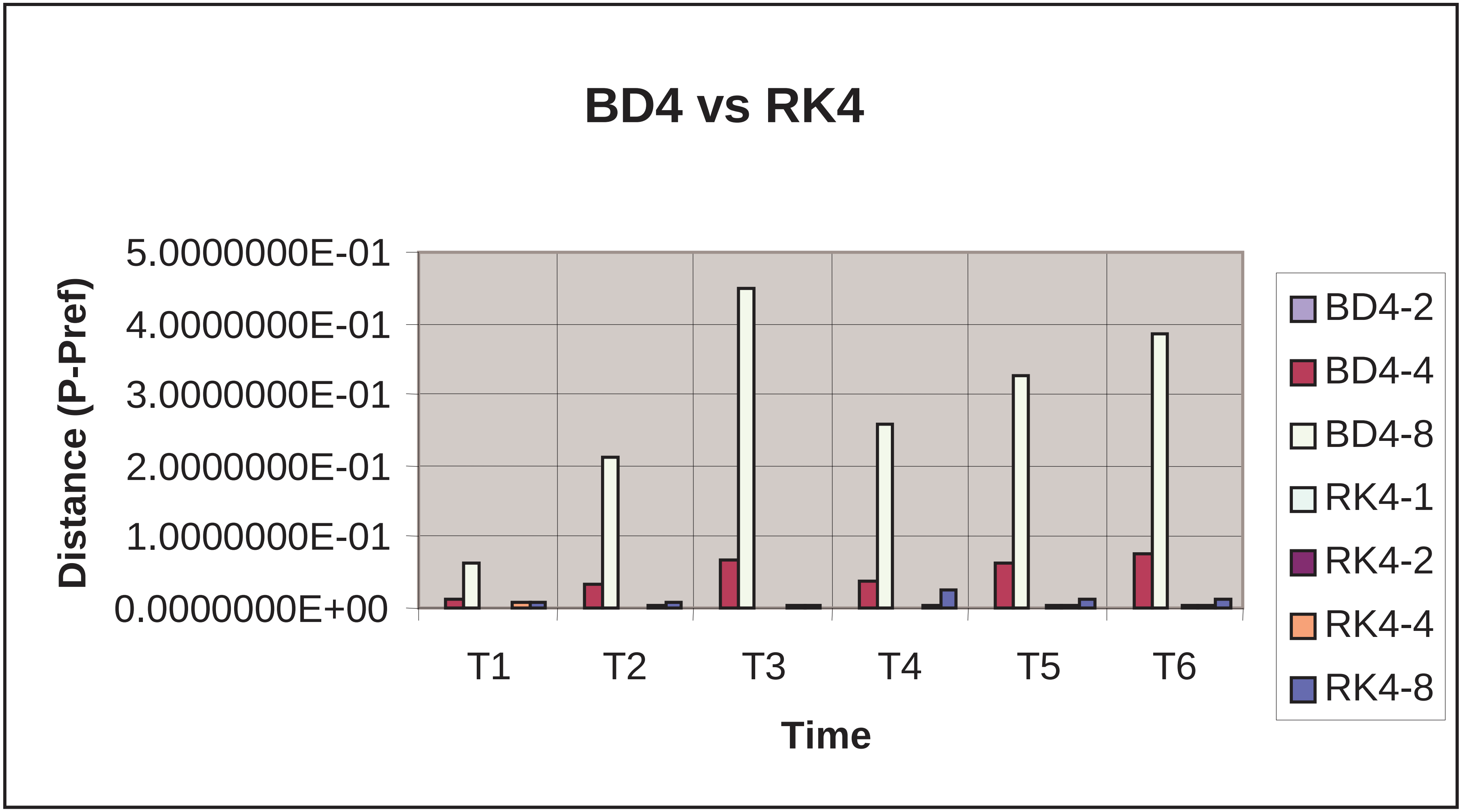
Point T4



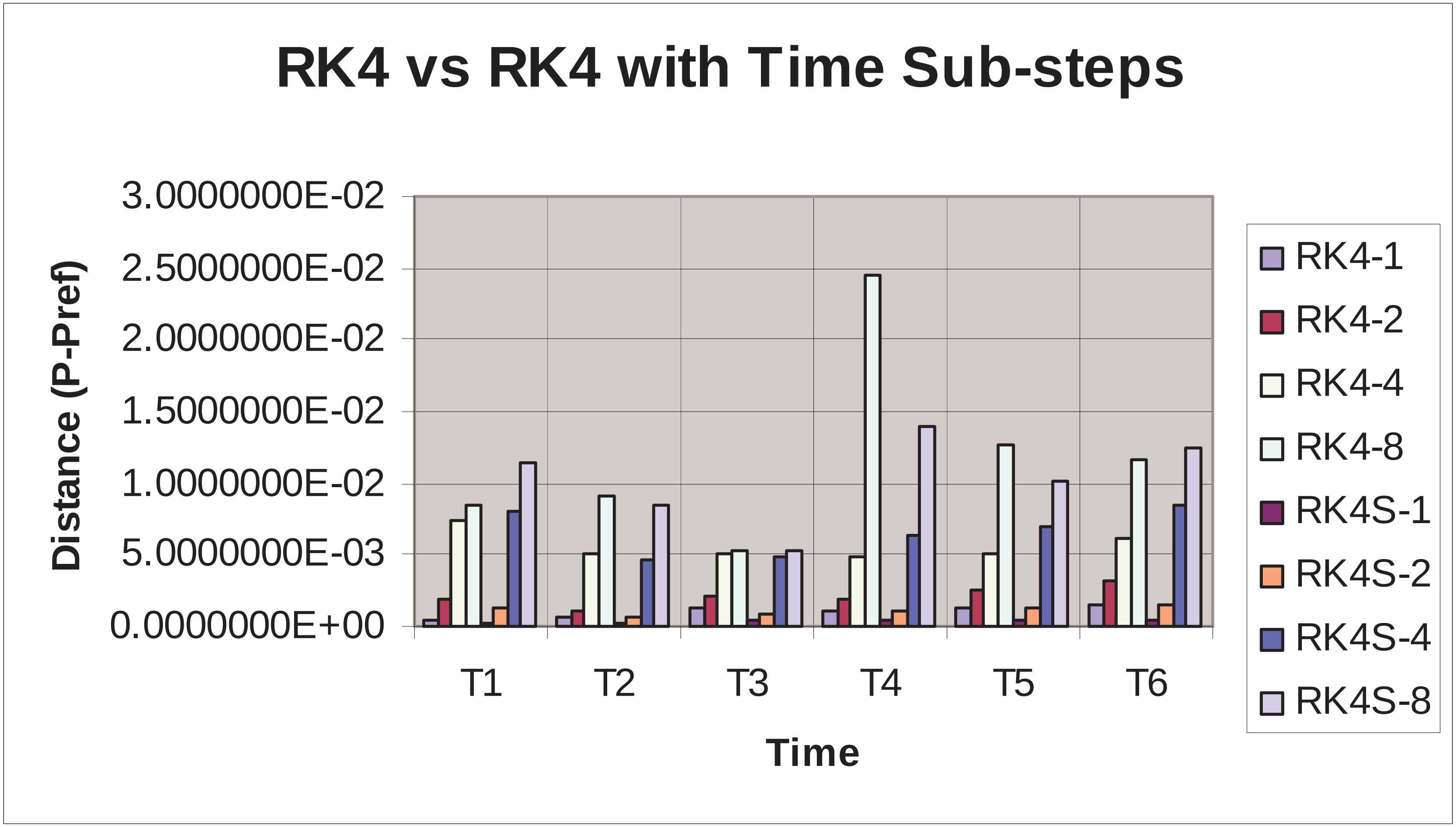
Point T5



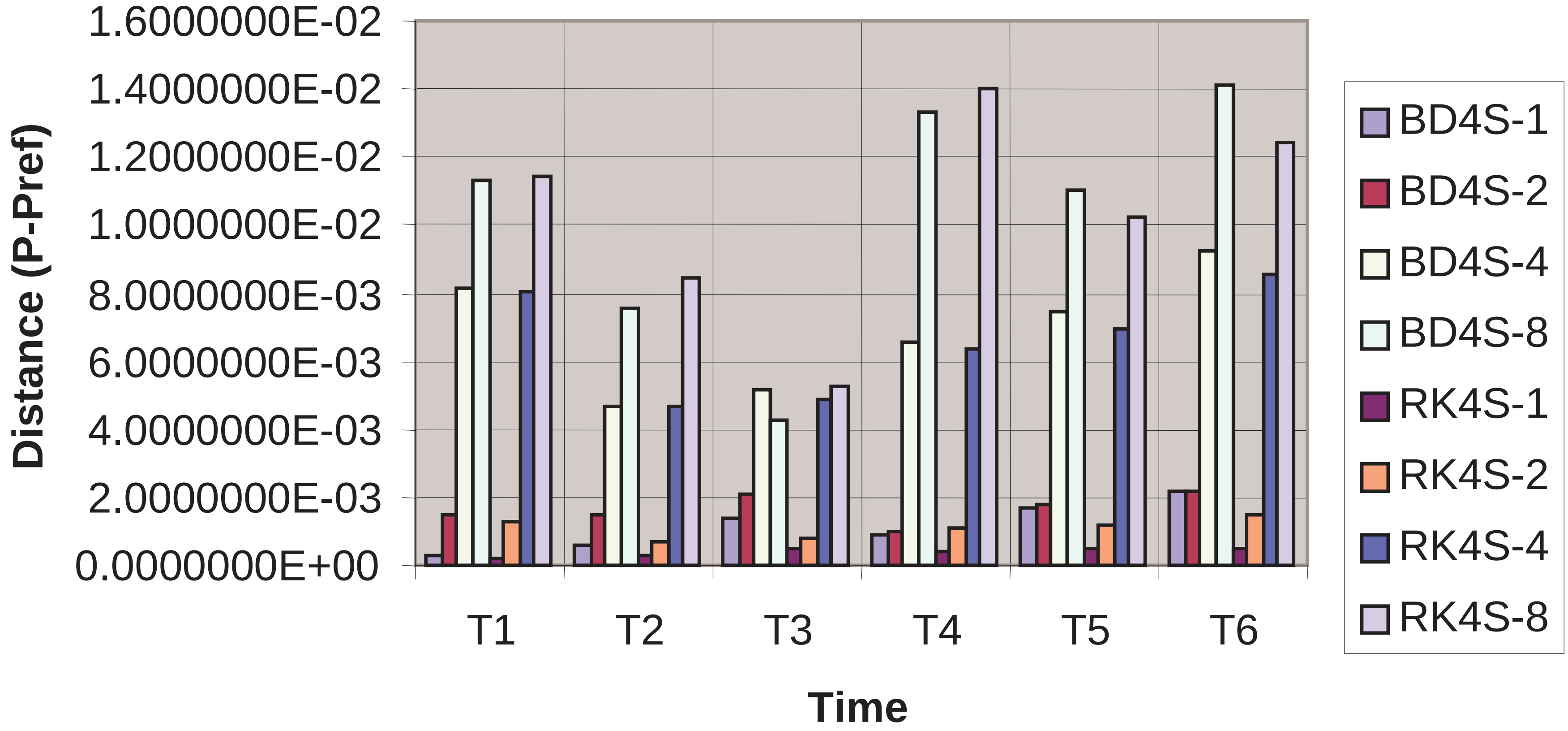
Point T6



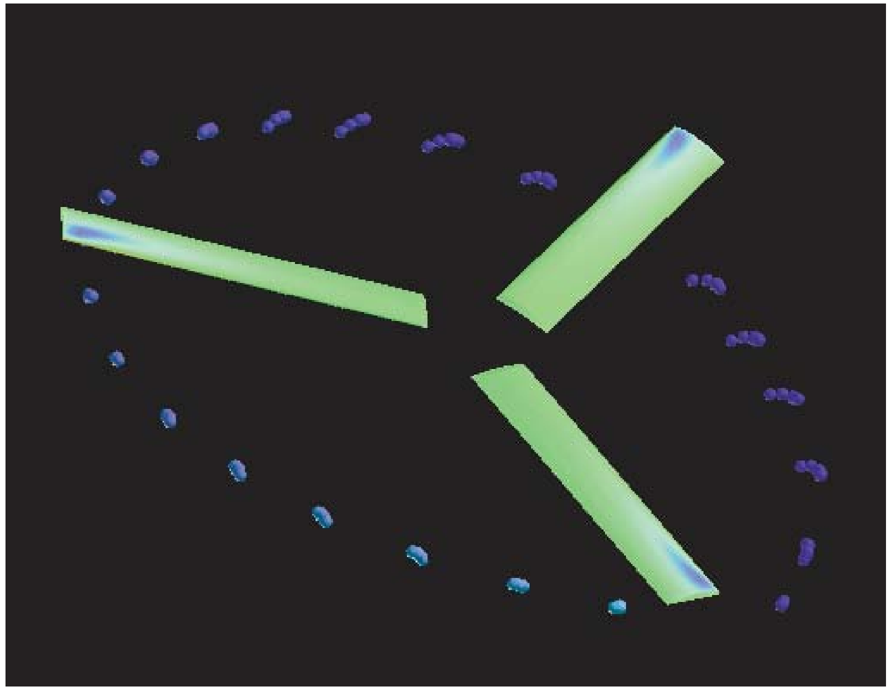
BD4 vs RK4
Distance (P-Pref)
5.0000000E-01
4.0000000E-01
3.0000000E-01
2.0000000E-01
1.0000000E-01
0.0000000E+00
T1
T2
T3
T4
T5
T6
Time
BD4-2
BD4-4
BD4-8
RK4-1
RK4-2
RK4-4
RK4-8


RK4 vs RK4 with Time Sub-steps
Distance (P-Pref)
3.0000000E-02
2.5000000E-02
2.0000000E-02
1.5000000E-02
1.0000000E-02
5.0000000E-03
0.0000000E+00
T1
T2
T3
T4
T5
T6
Time
RK4-1
RK4-2
RK4-4
RK4-8
RK4S-1
RK4S-2
RK4S-4
RK4S-8

BD4 vs RK4 (Both with Time Sub-steps)



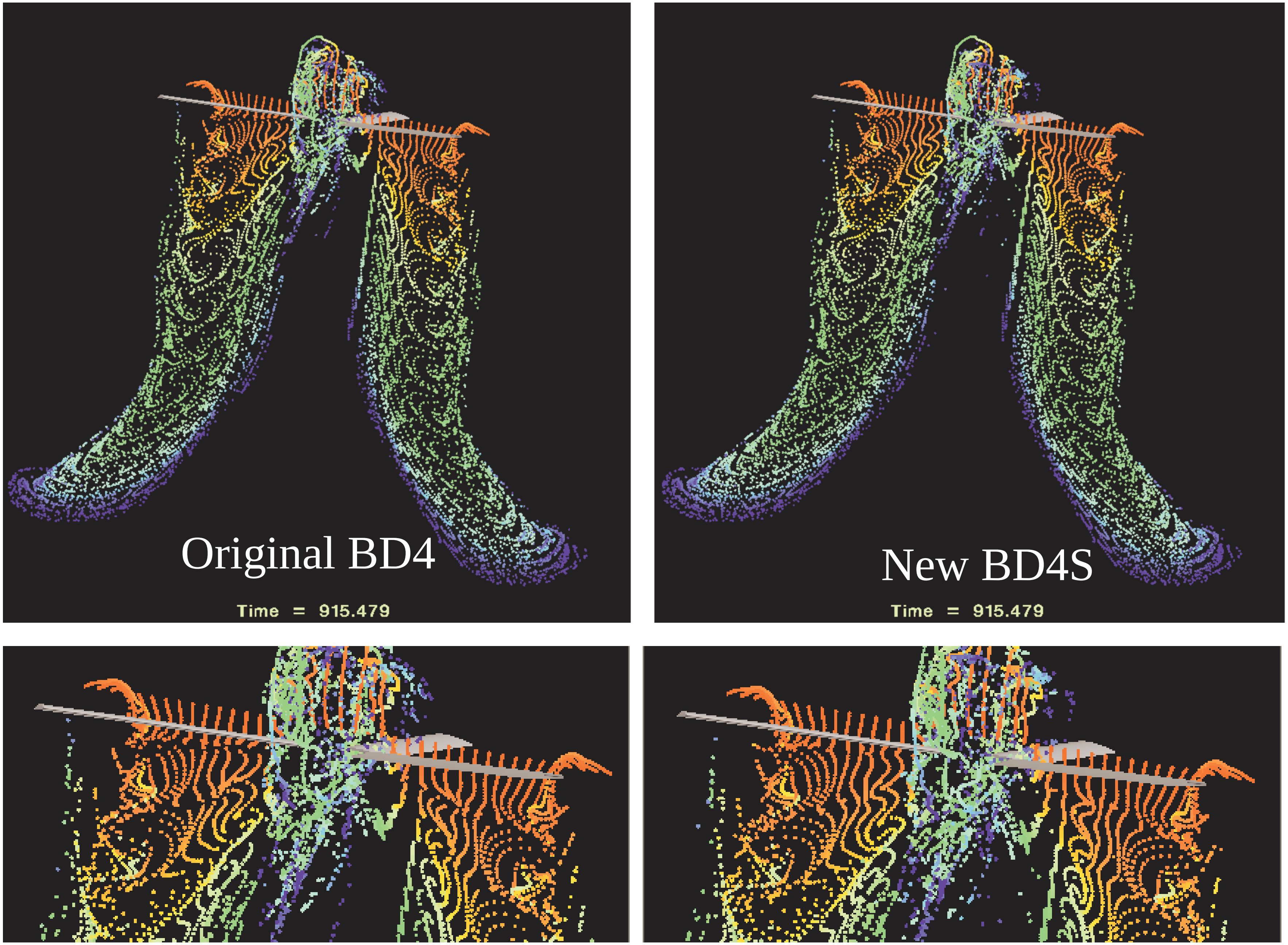
Streakline Model Problem 2
Rotor at Hover



- ❑ 160 Grids
- ❑ 72 Time Steps (5 degree rotation per time-step)
- ❑3 Bladed Helicopter System

Streaklines in Hover Simulation

- Release a line of streaklines (40) every time-step for 5 cycles
- 14400 particles released
- Original BD4 lost 457 particles
- New BD4S and RK4S lost 30 particles
- 15x reduction in lost particles


Original BD4
Time = 915.479
New BD4S
Time = 915.479

Conclusions

- Neither RK4 or BD4 with linear interpolated time sub-steps show 4th order behavior
  - BD4 without time sub-step interpolation shows largest error growth
- Applying time sub-steps has significant impact
  - Integration near bodies
  - Integration in off-body
- Number of lost particles in streakline calculation can be significantly reduced by using time sub-steps, improved 3D interpolation and wall modeling

Summary

- What have we learned?
  - Significant improvements can be made to visualization techniques when using rotorcraft domain knowledge
    - Speedup in visualization techniques
    - Reduce amount of data to be read
    - Sharing of Solver Knowledge with Post Processing
  - Particles loss in Streakline calculation is impacted:
    - Integration Scheme
    - Streakline integration time step
    - Temporal and Spatial Interpolation
    - Wall modeling